Projet en regression

Amal REKIK / Maa Eunice LAMAH

2023-10-16

Aperçu du jeu de données:

```
head(Boston)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21
##   medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4
## 5 36.2
## 6 28.7
```

Description des variables du jeu de données Boston

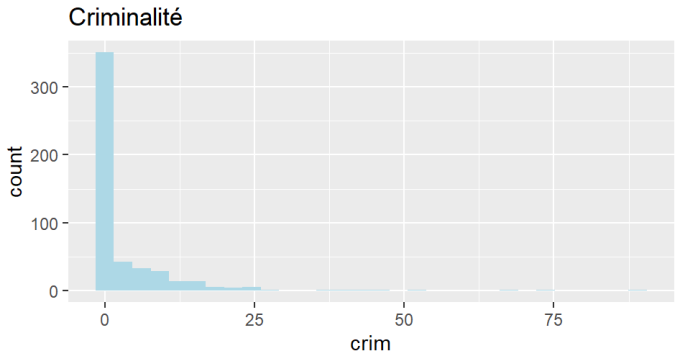
contient des informations sur le logement dans la région de Boston

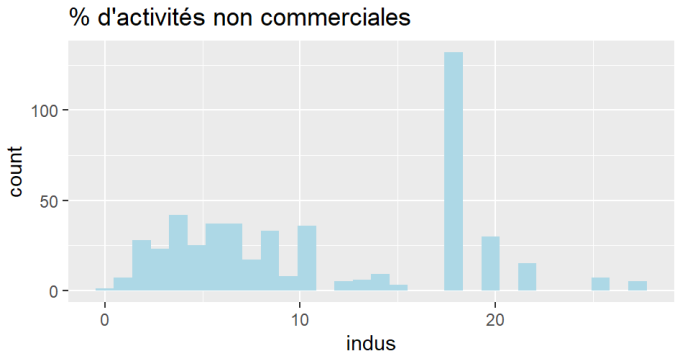
- **crim**: Taux de criminalité par habitant par ville.
- **zn**: Proportion de terrains résidentiels zonés pour des lots de plus de 25 000 pieds carrés.
- **indus**: Proportion d'acres d'activités non commerciales par ville.
- **chas**: Variable fictive pour Charles River (1 si le tronçon est lié à la rivière; 0 sinon).
- **nox**: Concentration en oxydes nitriques (parties pour 10 millions).
- **rm**: Nombre moyen de pièces par habitation.
- **age**: Proportion de logements occupés construits avant 1940.
- **dis**: Poids moyen des distances à cinq centres d'emploi de Boston.
- **rad**: Indice d'accessibilité aux routes radiales.
- **tax**: Taux d'imposition foncière à valeur totale pour 10 000 $.
- **ptratio**: Ratio élève-enseignant par ville.
- **black**: 1000(Bk - 0.63)^2 où Bk est la proportion d'Afro-Américains par ville.
- **lstat**: Pourcentage de la population avec un statut socio-économique inférieur.
- **medv**: Valeur médiane des logements occupés par leur propriétaire en milliers de dollars.

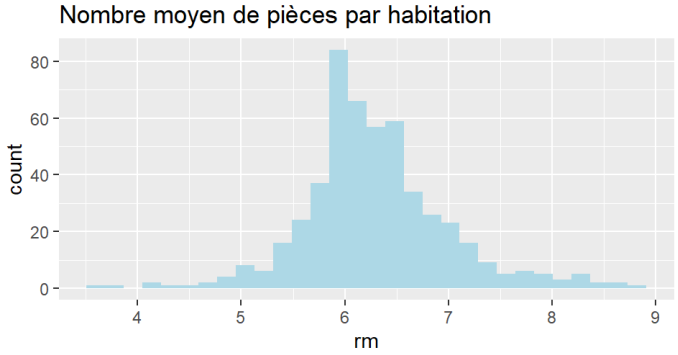
La correlation entre les variables

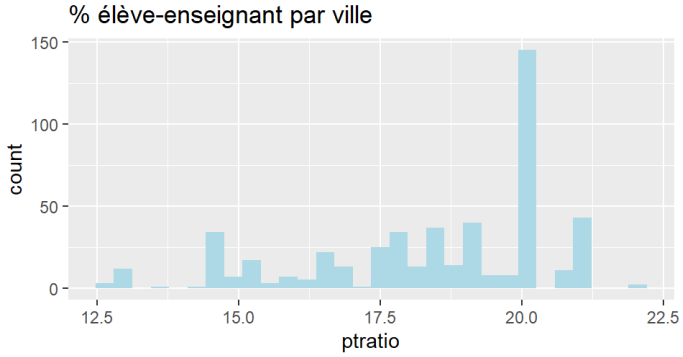
```
##               crim          zn       indus         chas         nox
## crim     1.00000000 -0.20046922  0.40658341 -0.055891582  0.42097171
## zn      -0.20046922  1.00000000 -0.53382819 -0.042696719 -0.51660371
## indus    0.40658341 -0.53382819  1.00000000  0.062938027  0.76365145
## chas    -0.05589158 -0.04269672  0.06293803  1.000000000  0.09120281
## nox      0.42097171 -0.51660371  0.76365145  0.091202807  1.00000000
## rm      -0.21924670  0.31199059 -0.39167585  0.091251225 -0.30218819
## age      0.35273425 -0.56953734  0.64477851  0.086517774  0.73147010
## dis     -0.37967009  0.66440822 -0.70802699 -0.099175780 -0.76923011
## rad      0.62550515 -0.31194783  0.59512927 -0.007368241  0.61144056
## tax      0.58276431 -0.31456332  0.72076018 -0.035586518  0.66802320
## ptratio  0.28994558 -0.39167855  0.38324756 -0.121515174  0.18893268
## black   -0.38506394  0.17552032 -0.35697654  0.048788485 -0.38005064
## lstat    0.45562148 -0.41299457  0.60379972 -0.053929298  0.59087892
## medv    -0.38830461  0.36044534 -0.48372516  0.175260177 -0.42732077
##                 rm         age         dis          rad         tax    ptratio
## crim    -0.21924670  0.35273425 -0.37967009  0.625505145  0.58276431  0.2899456
## zn       0.31199059 -0.56953734  0.66440822 -0.311947826 -0.31456332 -0.3916785
## indus   -0.39167585  0.64477851 -0.70802699  0.595129275  0.72076018  0.3832476
## chas     0.09125123  0.08651777 -0.09917578 -0.007368241 -0.03558652 -0.1215152
## nox     -0.30218819  0.73147010 -0.76923011  0.611440563  0.66802320  0.1889327
## rm       1.00000000 -0.24026493  0.20524621 -0.209846668 -0.29204783 -0.3555015
## age     -0.24026493  1.00000000 -0.74788054  0.456022452  0.50645559  0.2615150
## dis      0.20524621 -0.74788054  1.00000000 -0.494587930 -0.53443158 -0.2324705
## rad     -0.20984667  0.45602245 -0.49458793  1.000000000  0.91022819  0.4647412
## tax     -0.29204783  0.50645559 -0.53443158  0.910228189  1.00000000  0.4608530
## ptratio -0.35550149  0.26151501 -0.23247054  0.464741179  0.46085304  1.0000000
## black    0.12806864 -0.27353398  0.29151167 -0.444412816 -0.44180801 -0.1773833
## lstat   -0.61380827  0.60233853 -0.49699583  0.488676335  0.54399341  0.3740443
## medv     0.69535995 -0.37695457  0.24992873 -0.381626231 -0.46853593 -0.5077867
##               black      lstat       medv
## crim    -0.38506394  0.4556215 -0.3883046
## zn       0.17552032 -0.4129946  0.3604453
## indus   -0.35697654  0.6037997 -0.4837252
## chas     0.04878848 -0.0539293  0.1752602
## nox     -0.38005064  0.5908789 -0.4273208
## rm       0.12806864 -0.6138083  0.6953599
## age     -0.27353398  0.6023385 -0.3769546
## dis      0.29151167 -0.4969958  0.2499287
## rad     -0.44441282  0.4886763 -0.3816262
## tax     -0.44180801  0.5439934 -0.4685359
## ptratio -0.17738330  0.3740443 -0.5077867
## black    1.00000000 -0.3660869  0.3334608
## lstat   -0.36608690  1.0000000 -0.7376627
## medv     0.33346082 -0.7376627  1.0000000
```

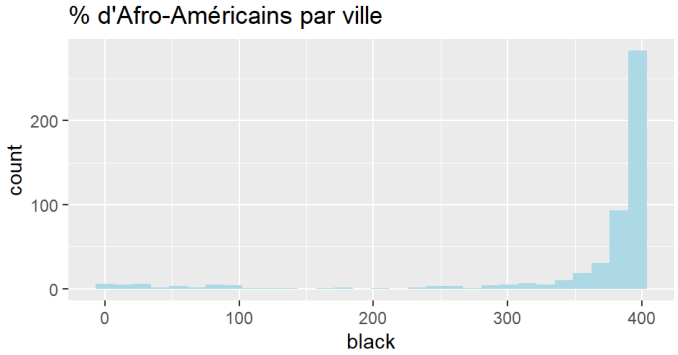
La distribution des variables

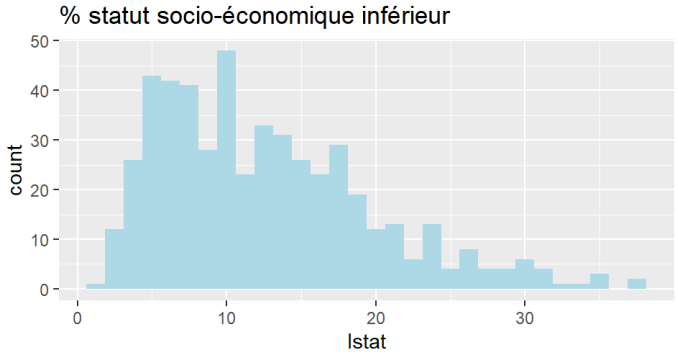












Regression multiple

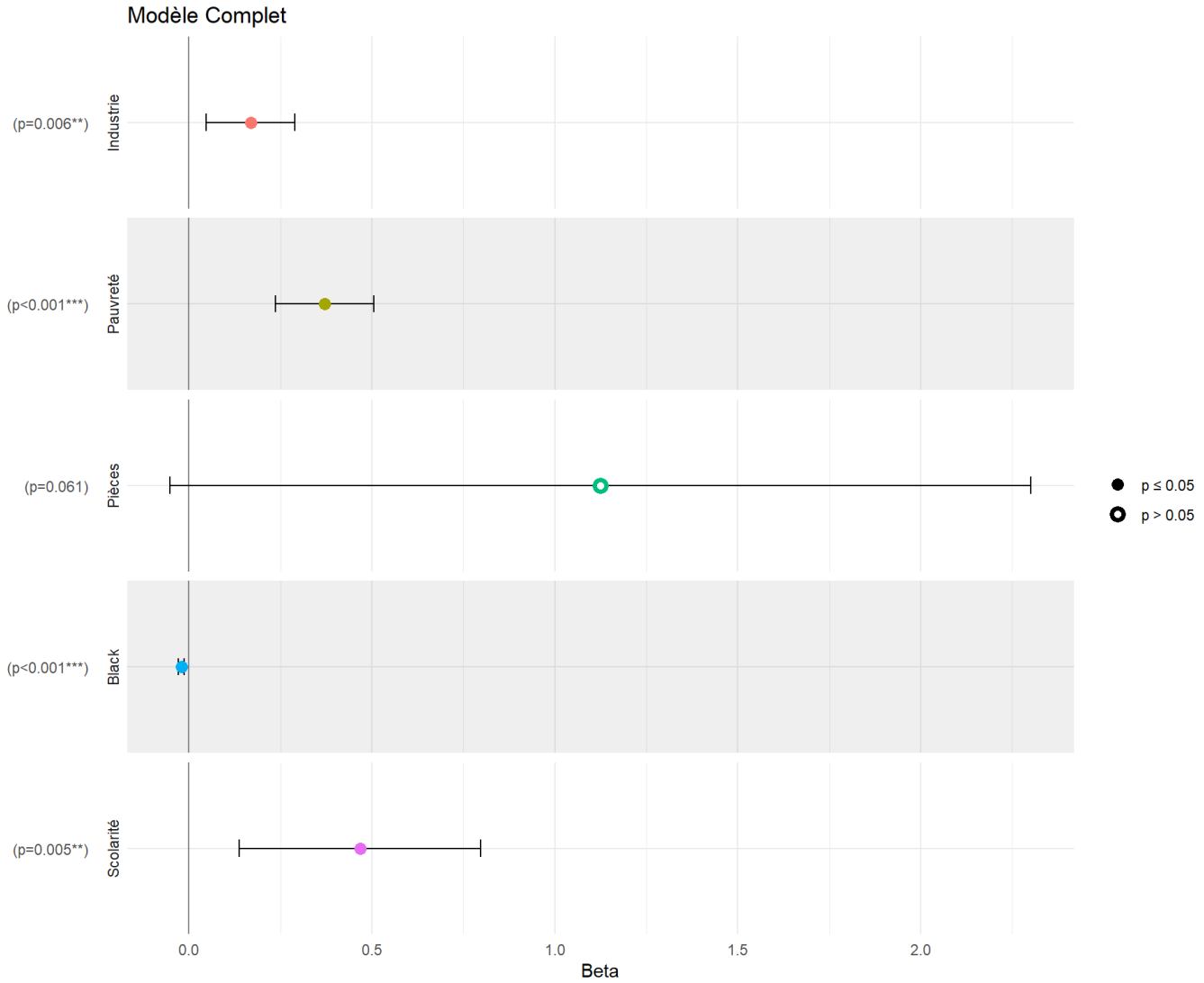
Renommer les variables

```
reg <- lm(crim ~ indus + lstat + rm + black + ptratio, data=reg_data)
summary(reg)
```

```
##
## Call:
## lm(formula = crim ~ indus + lstat + rm + black + ptratio, data = reg_data)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -12.730  -2.492  -0.502   1.167  81.727
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -11.427671   5.949093  -1.921  0.05531 .
## indus         0.169012   0.061575   2.745  0.00627 **
## lstat         0.370849   0.068563   5.409 9.85e-08 ***
## rm            1.125077   0.598607   1.879  0.06076 .
## black        -0.020270   0.003916  -5.175 3.30e-07 ***
## ptratio       0.467365   0.167497   2.790  0.00547 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.274 on 500 degrees of freedom
## Multiple R-squared:  0.2919, Adjusted R-squared:  0.2848
## F-statistic: 41.22 on 5 and 500 DF,  p-value: < 2.2e-16
```

Representation des coefficients

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



Analyse de la régression linéaire

Nous avons effectué une régression linéaire pour examiner les facteurs influençant le taux de criminalité à Boston. Voici les variables utilisées :

- `crim` : Taux de criminalité par habitant.
- Prédicteurs: `indus` , `lstat` , `rm` , `black` , et `ptratio` .

Interprétation des coefficients

- `(Intercept)` : L'ordonnée à l'origine est estimée à -11.43. Cela représente le taux de criminalité attendu lorsque toutes les variables explicatives sont égales à zéro.
- `indus` : Pour chaque augmentation unitaire de `indus` , le `crim` augmente en moyenne de 0.169.
- `lstat` : Pour chaque augmentation de 1% de `lstat` , le `crim` augmente en moyenne de 0.371.
- `rm` : Pour chaque pièce supplémentaire, le `crim` augmente en moyenne de 1.125.
- `black` : Pour chaque augmentation unitaire de `black` , le `crim` diminue en moyenne de 0.0203.
- `ptratio` : Pour chaque augmentation unitaire du `ptratio` , le `crim` augmente en moyenne de 0.467.

Significativité des coefficients

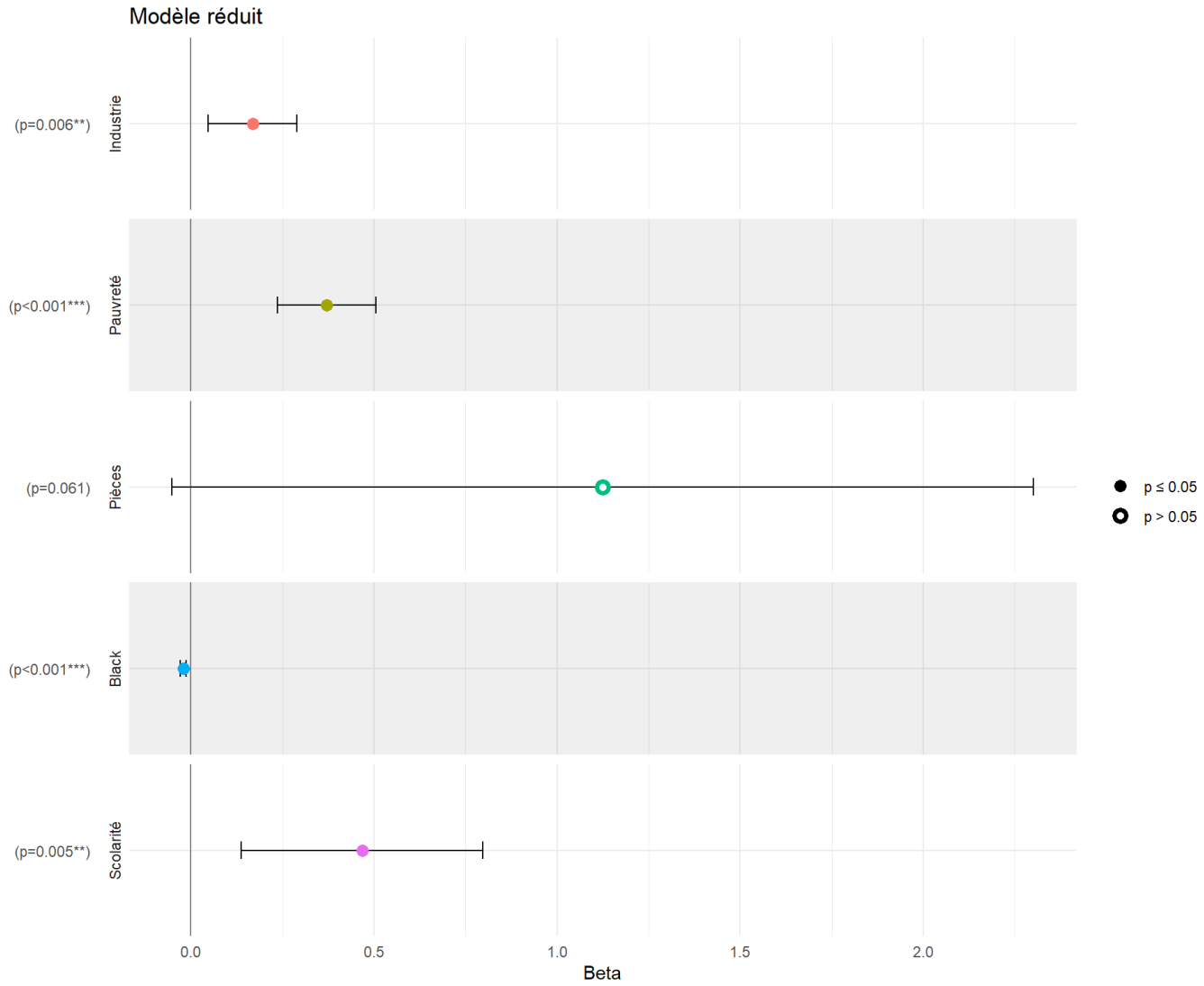
- `indus` , `lstat` , `black` , et `ptratio` sont significatives au seuil de 1%.
- `rm` est marginale avec une p-valeur juste supérieure à 0.05.

Mesures de qualité de l'ajustement

- Le R^2 multiple est de 0.2919, signifiant que près de 29.19% de la variabilité de `crim` est expliquée par le modèle.
- Le R^2 ajusté est de 0.2848.
- La statistique F est très significative, ce qui veut dire que le modèle est utile pour prédire `crim` .

Conclusions

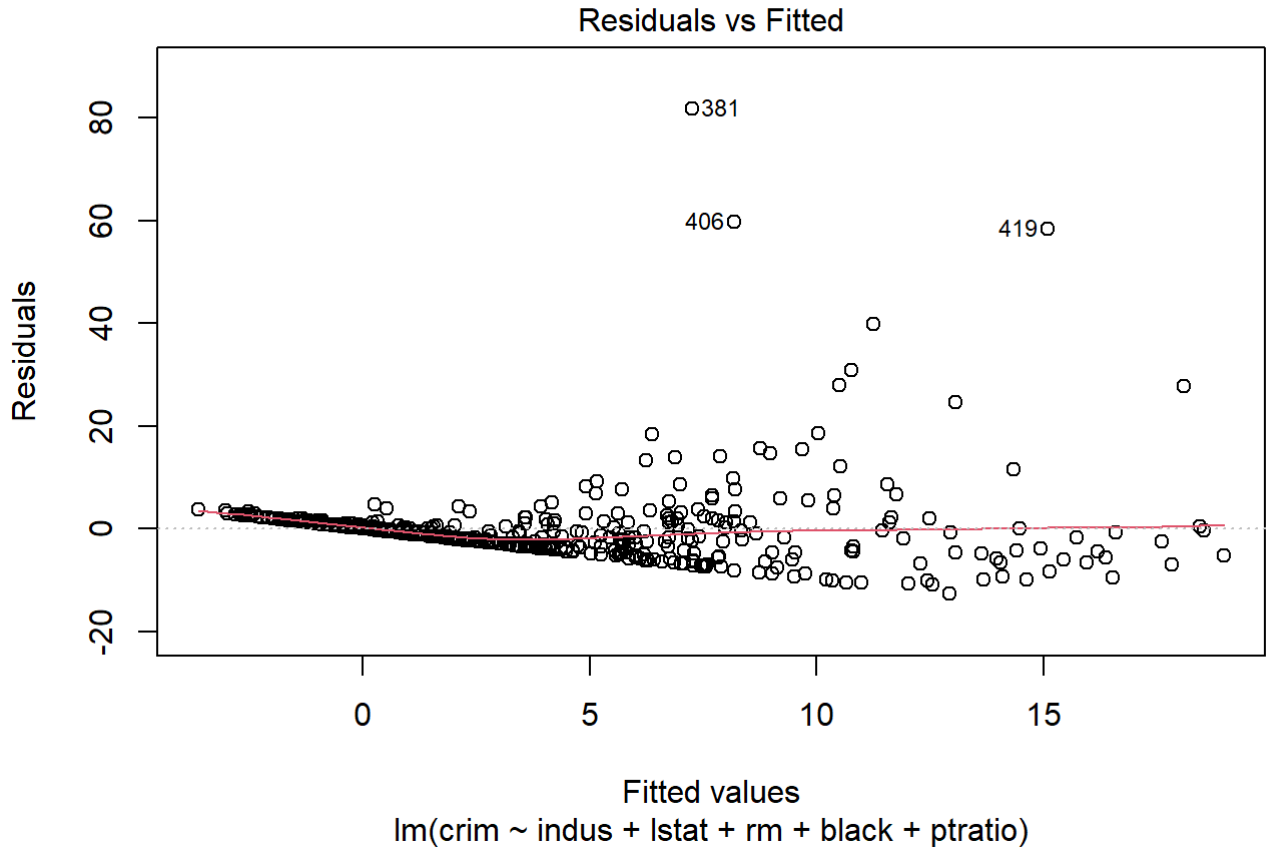
Le modèle fournit des insights pertinents sur la relation entre le taux de criminalité et les variables qu'on a choisies. Cependant, une grande variabilité dans `crim` n'est pas expliquée, il faudrait donc explorer d'autres variables ou interactions.



Hétéroscédasticité :

Visualisation des résidus par rapport aux valeurs prédites.

```
plot(reg, 1)
```



- Les résidus montrent une forme d'éventail, cela indique de l'hétéroscédasticité, ce qui est une violation de l'une des hypothèses de la régression linéaire.

Test de Breusch-Pagan pour l'hétéroscédasticité.

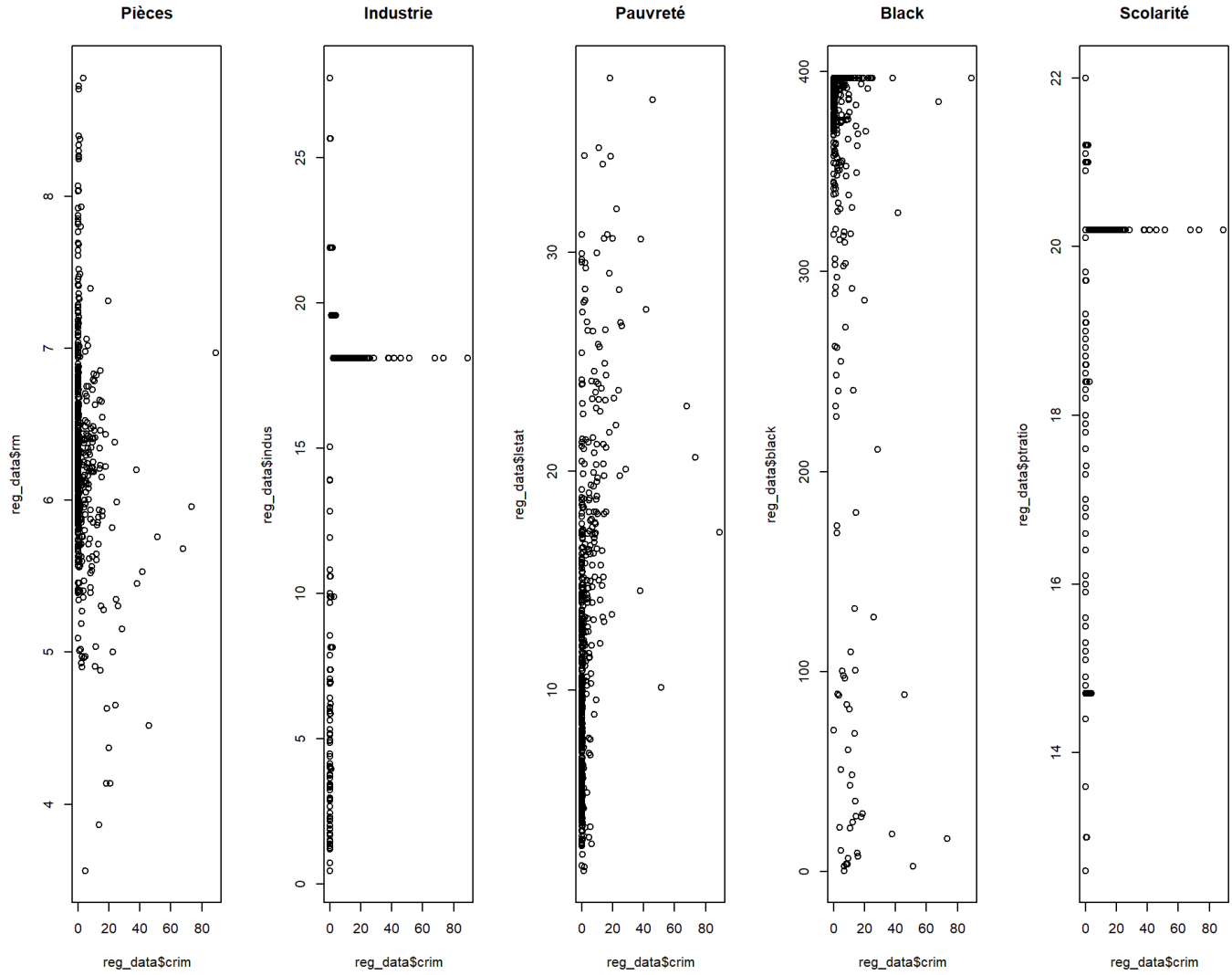
```
bptest(reg)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  reg
## BP = 16.642, df = 5, p-value = 0.005231
```

Conclusion

- L'analyse des résidus par rapport aux valeurs ajustées a révélé des signes d'hétéroscédasticité, ce qui a est confirmé par le test de Breusch-Pagan.
- L'hétéroscédasticité signifie que la variance des erreurs du modèle n'est pas constante. Cette situation peut biaiser les estimations des erreurs standards et, par conséquent, les tests d'hypothèses sur les coefficients de régression.
- Il serait donc pertinent d'envisager des méthodes pour aborder cette hétéroscédasticité, comme transformer certaines variables ou utiliser des erreurs standards robustes.

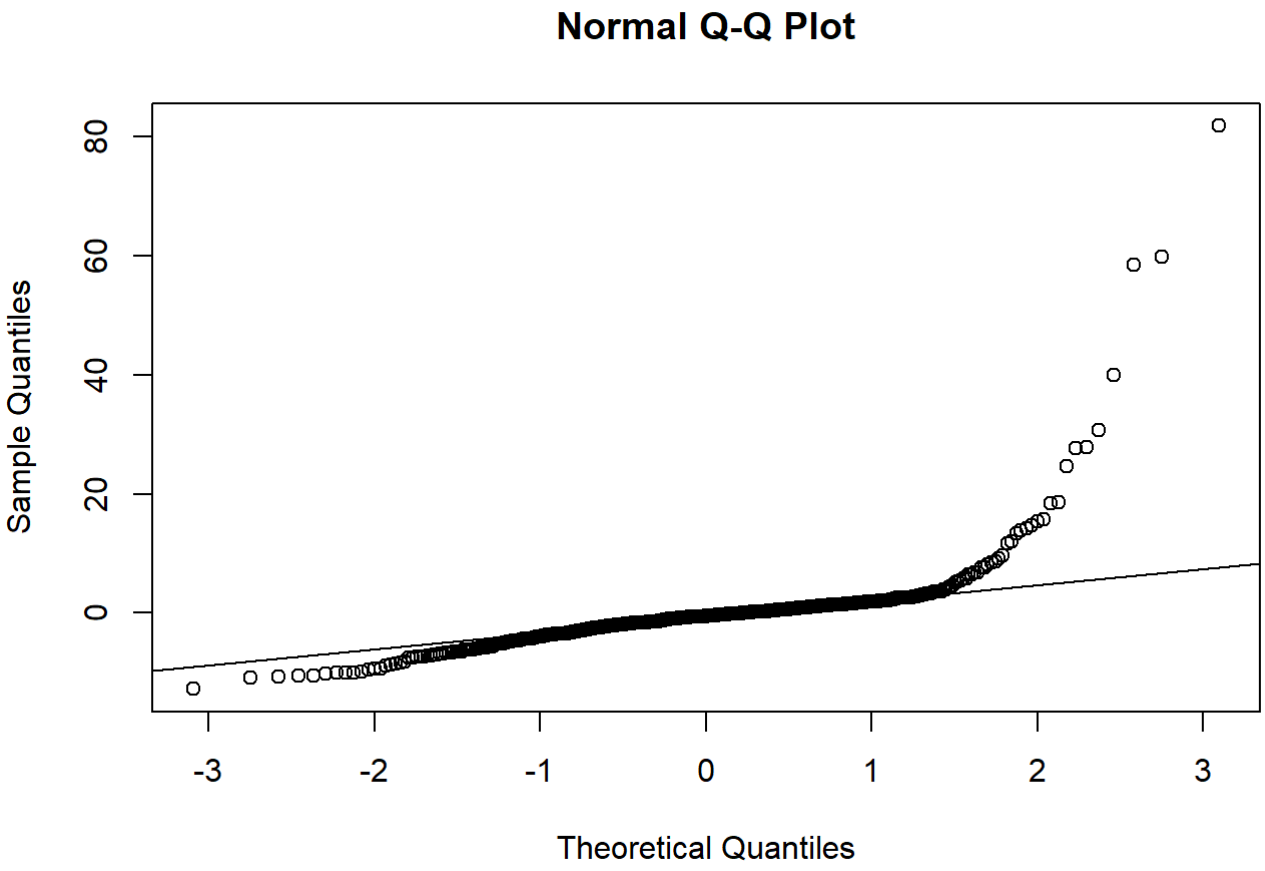
vérification de la relation linéaire des variables



Normalité des Résidus

```
residu <- residuals(reg)

qqnorm(residu)
qqline(residu)
```



Test de vérification

```
shapiro.test(residu)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residu
## W = 0.55201, p-value < 2.2e-16
```

```
mean(residu)
```

```
## [1] 5.998484e-16
```

Nos résidus ne suivent pas une distribution normale selon le test, ce qui veut dire que notre modèle peut ne pas capturer complètement la complexité des données.

Absence d'Autocorrélation

```
dwtest(reg)
```

```
##
##  Durbin-Watson test
##
## data:  reg
## DW = 1.2554, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

Interpretaion

Autocorrélation des résidus : Le test de Durbin-Watson a révélé une autocorrélation positive significative dans les résidus de notre modèle (`DW = 1.2554, p-value < 2.2e-16`). Cela indique que nos résidus ne sont pas indépendants, ce qui suggère la présence de certaines tendances temporelles ou de structures non capturées par notre modèle.

Absence de Colinéarité

```
library(car)
```

```
## Le chargement a nécessité le package : carData
```

```
vif(reg)
```

```
##    indus    lstat       rm    black  ptratio
## 1.702982 2.287810 1.688236 1.220114 1.254935
```

Nous avons évalué la multicolinéarité dans notre modèle de régression à l'aide du Variance Inflation Factor (VIF). Le VIF mesure l'ampleur de l'inflation de la variance dans un modèle de régression multiple, indiquant une forte corrélation entre une variable indépendante et les autres variables indépendantes.

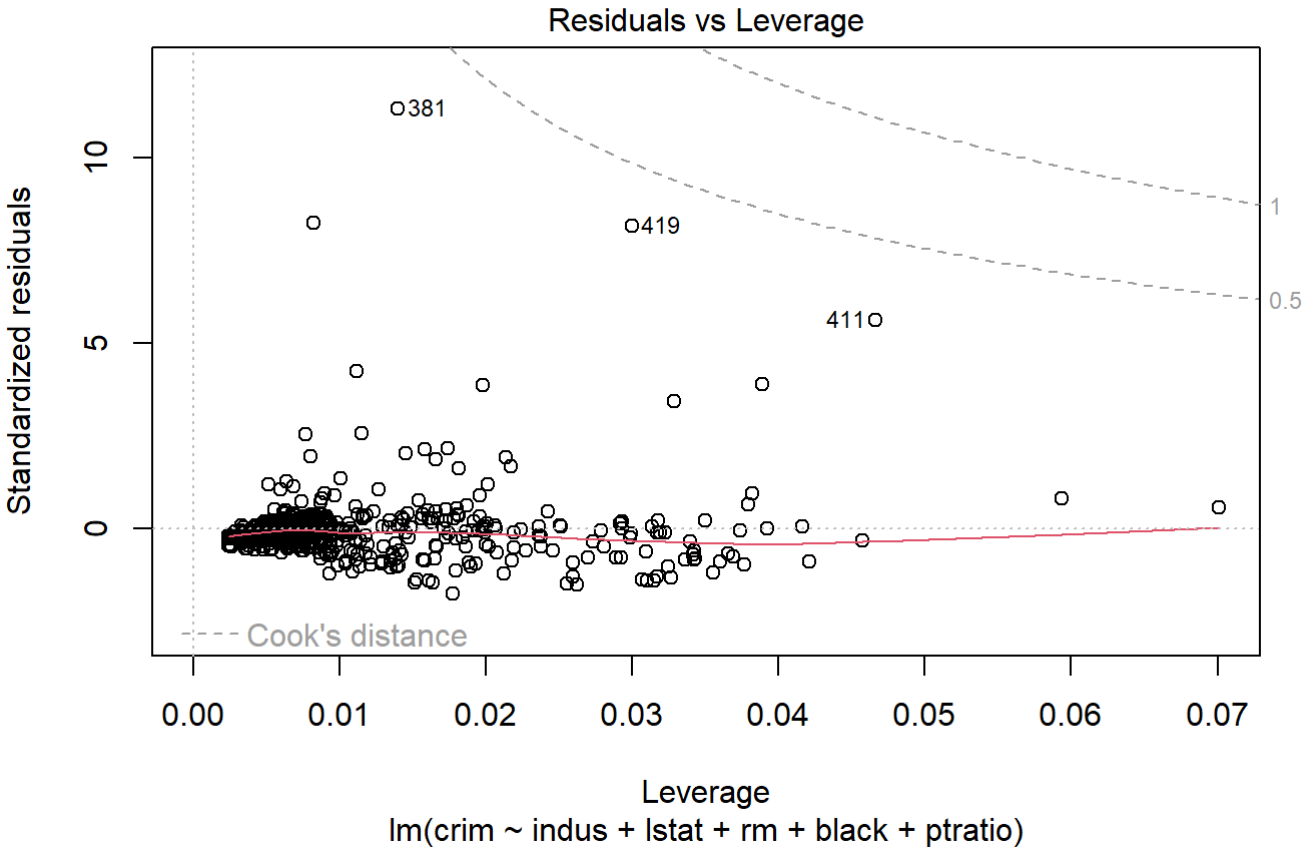
Les valeurs VIF pour nos variables explicatives sont les suivantes :

- `indus` : $VIF = 1.70$
- `lstat` : $VIF = 2.29$
- `rm` : $VIF = 1.69$
- `black` : $VIF = 1.22$
- `ptratio` : $VIF = 1.25$

Toutes les valeurs VIF sont inférieures à 5, ce qui indique qu'il n'y a pas de problème majeur d'inflation de la variance dans notre modèle. Cela suggère que nos variables explicatives ne sont pas fortement corrélées entre elles, ce qui renforce la stabilité de notre modèle de régression.

Graphiques de levier et de distance de Cook.

```
plot(reg, which=5)  # Graphique de distance de Cook
```



Analyse des points influents

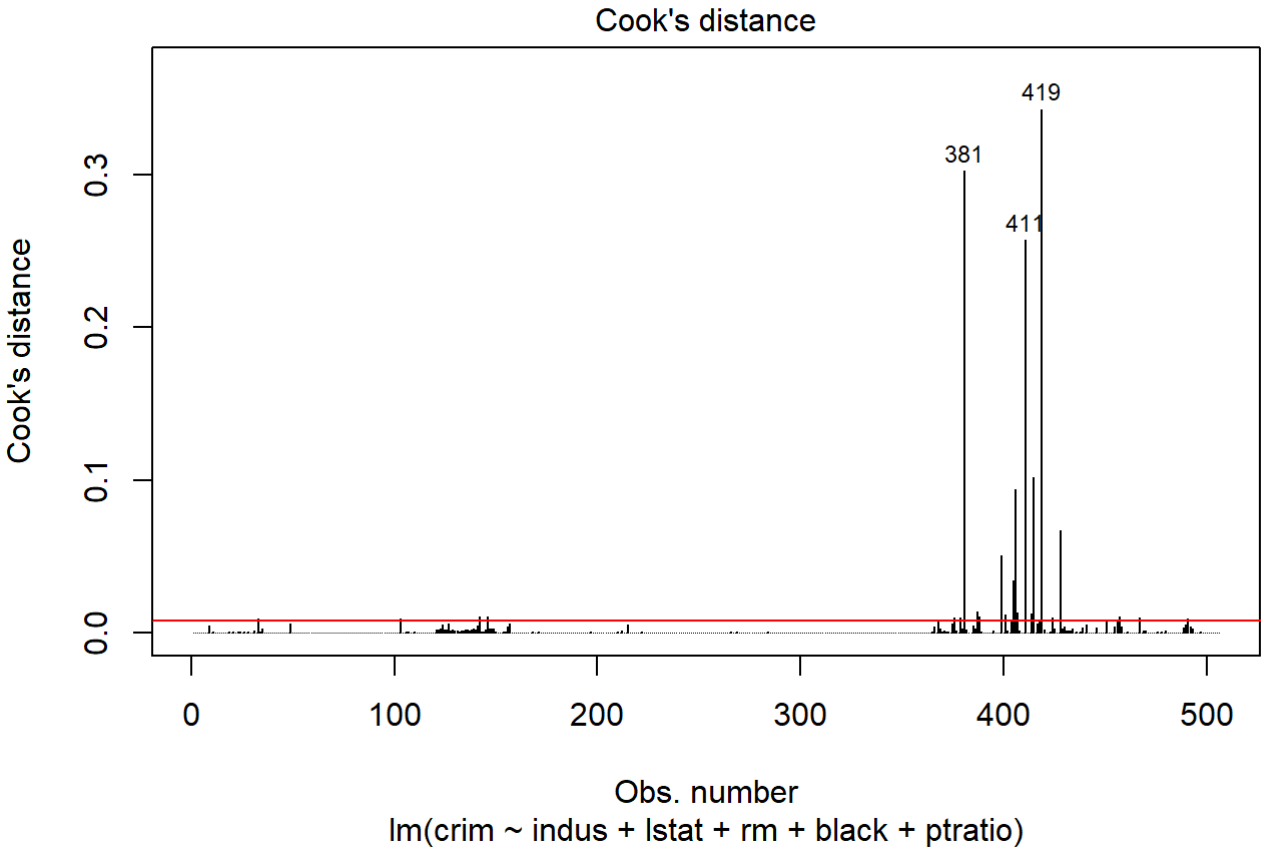
En examinant les distances de Cook, nous avons identifié plusieurs observations qui pourraient avoir une influence substantielle sur notre modèle. Les observations 381, 411, et 419 ont montré des distances de Cook particulièrement élevées.

Cela pourrait être dû à des valeurs aberrantes ou à d'autres particularités de ces observations. Voyons de plus près (avec la 1ere obs qui va servir de comparaison)

```
cook_data = Boston[c(381,411,419,1),]
cook_data
```

```
##         crim zn indus chas   nox    rm   age    dis rad tax ptratio  black
## 381 88.97620  0 18.10    0 0.671 6.968  91.9 1.4165  24 666    20.2 396.90
## 411 51.13580  0 18.10    0 0.597 5.757 100.0 1.4130  24 666    20.2   2.60
## 419 73.53410  0 18.10    0 0.679 5.957 100.0 1.8026  24 666    20.2  16.45
## 1    0.00632 18  2.31    0 0.538 6.575  65.2 4.0900   1 296    15.3 396.90
##     lstat medv
## 381 17.21 10.4
## 411 10.11 15.0
## 419 20.62  8.8
## 1    4.98 24.0
```

```
# Visualiser les points influents
plot(reg, which=4)
abline(h = 4/((length(Boston$crim) - length(coef(reg)) - 2)), col="red")  # Ligne de référenc
e
```



Régression linéaire sur le nouveau modèle

```
# Supprimer les observations influentes
data_sans <- reg_data[-c(381, 411, 419), ]

# Refaire la régression
reg_sans <- lm(crim ~ indus + lstat + rm + black + ptratio, data = data_sans)

# Afficher un résumé du nouveau modèle
summary(reg_sans)
```

```
##
## Call:
## lm(formula = crim ~ indus + lstat + rm + black + ptratio, data = data_sans)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -12.045  -2.187  -0.319   1.293  59.940
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10.496372   4.453323  -2.357  0.01881 *
## indus         0.139560   0.045644   3.058  0.00235 **
## lstat         0.385953   0.051210   7.537 2.30e-13 ***
## rm            0.820540   0.446345   1.838  0.06661 .
## black        -0.012943   0.002998  -4.317 1.91e-05 ***
## ptratio       0.366416   0.124123   2.952  0.00331 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.386 on 497 degrees of freedom
## Multiple R-squared:  0.369,  Adjusted R-squared:  0.3626
## F-statistic: 58.12 on 5 and 497 DF,  p-value: < 2.2e-16
```

Comparaison du modèle initial vs Modèle sans observations influentes

Intercept:

- Modèle initial : -11.427671 (p = 0.05531)
- Modèle sans observations influentes : -10.496372 (p = 0.01881)

L'ordonnée à l'origine est légèrement modifiée après avoir retiré les observations influentes, devenant plus significative dans le second modèle.

indus:

- Modèle initial : 0.169012 (p = 0.00627)
- Modèle sans observations influentes : 0.139560 (p = 0.00235)

La suppression des observations influentes a légèrement réduit la taille de l'effet de `indus` mais a rendu cette variable plus significative.

lstat:

- Modèle initial : 0.370849 (p < 0.001)
- Modèle sans observations influentes : 0.385953 (p < 0.001)

`lstat` reste significative dans les deux modèles, mais son effet a augmenté légèrement après avoir retiré les observations influentes.

rm:

- Modèle initial : 1.125077 (p = 0.06076)
- Modèle sans observations influentes : 0.820540 (p = 0.06661)

L'effet de `rm` a diminué après suppression des observations influentes et reste marginalement significatif dans les deux modèles.

black:

- Modèle initial : -0.020270 ($p < 0.001$)
- Modèle sans observations influentes : -0.012943 ($p < 0.001$)

L'effet de `black` sur `crim` a diminué en magnitude mais reste hautement significatif dans les deux modèles.

ptratio:

- Modèle initial : 0.467365 ($p = 0.00547$)
- Modèle sans observations influentes : 0.366416 ($p = 0.00331$)

La suppression des observations influentes a légèrement réduit l'effet de `ptratio` mais a rendu cette variable plus significative.

Diagnostics du modèle:

Erreur standard des résidus:

- Modèle initial : 7.274
- Modèle sans observations influentes : 5.386

L'erreur standard des résidus a diminué de manière significative après suppression des observations influentes, ce qui suggère que le nouveau modèle a une meilleure précision dans ses prédictions.

R^2:

- Modèle initial : 0.2919
- Modèle sans observations influentes : 0.369

Le R^2 a augmenté, indiquant que le modèle sans observations influentes explique une plus grande proportion de la variance dans `crim` .

Valeur F:

- Modèle initial : 41.22
- Modèle sans observations influentes : 58.12

La statistique F a augmenté, suggérant que le modèle ajusté est plus significatif que le modèle initial.

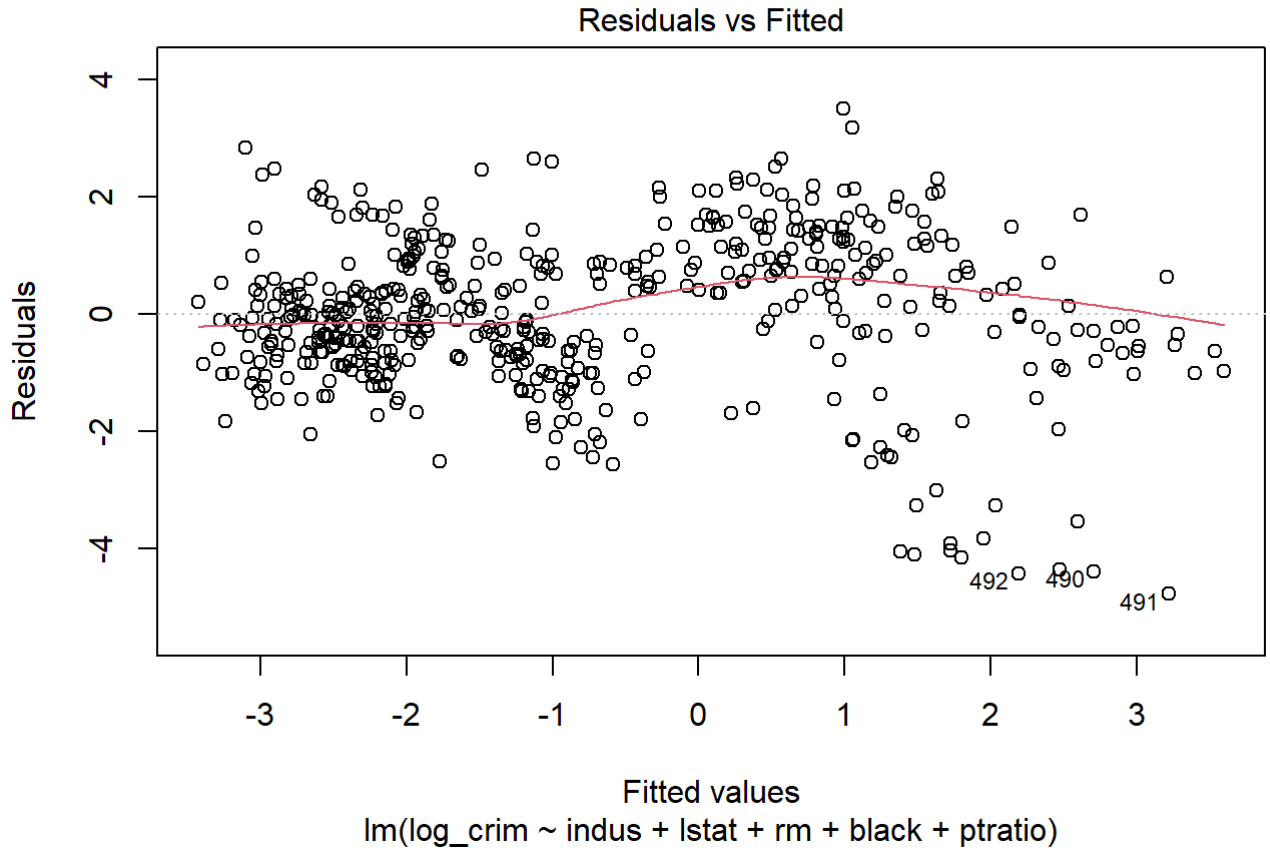
Testons avec le log de criminalité

```
reg_data$log_crim <- log(reg_data$crim)
reg_log <- lm(log_crim ~ indus + lstat + rm + black + ptratio, data = reg_data)
summary(reg_log)
```

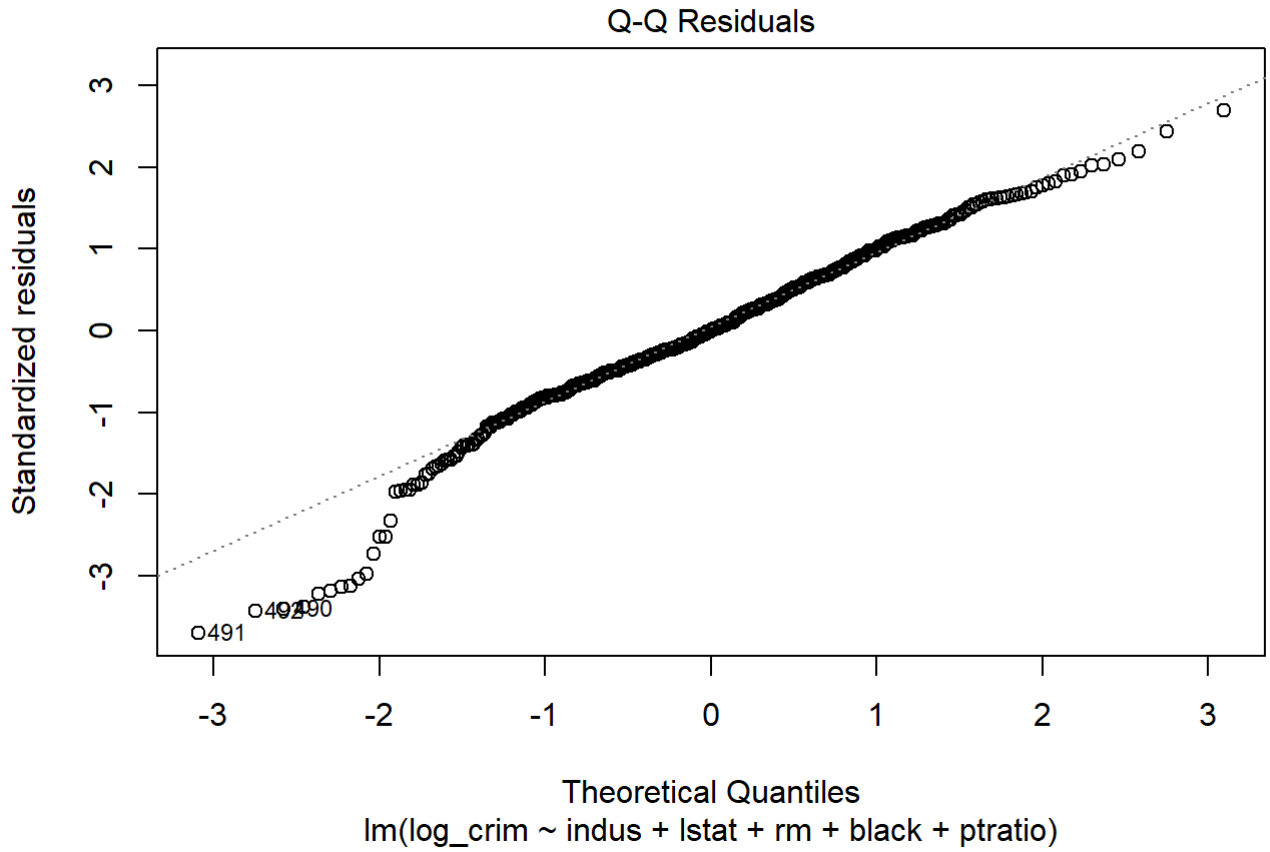
```
##
## Call:
## lm(formula = log_crim ~ indus + lstat + rm + black + ptratio,
##     data = reg_data)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -4.7858 -0.7318 -0.0052  0.8645  3.4988
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.4593488  1.0661769  -6.058 2.71e-09 ***
## indus        0.1554667  0.0110352  14.088  < 2e-16 ***
## lstat        0.0915022  0.0122877   7.447 4.22e-13 ***
## rm           0.4060944  0.1072804   3.785 0.000172 ***
## black       -0.0045255  0.0007019  -6.448 2.68e-10 ***
## ptratio      0.1003341  0.0300183   3.342 0.000893 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.304 on 500 degrees of freedom
## Multiple R-squared:   0.64,  Adjusted R-squared:  0.6364
## F-statistic: 177.8 on 5 and 500 DF,  p-value: < 2.2e-16
```

Ce modèle avec la variable dépendante transformée (log_crim) explique 64% de la variance dans log_crim. Chaque coefficient est significatif, ce qui suggère que toutes les variables choisies sont pertinentes pour expliquer la variation dans log_crim. La transformation logarithmique a aidé à linéariser les relations et à stabiliser la variance des résidus, conduisant à un meilleur ajustement du modèle.

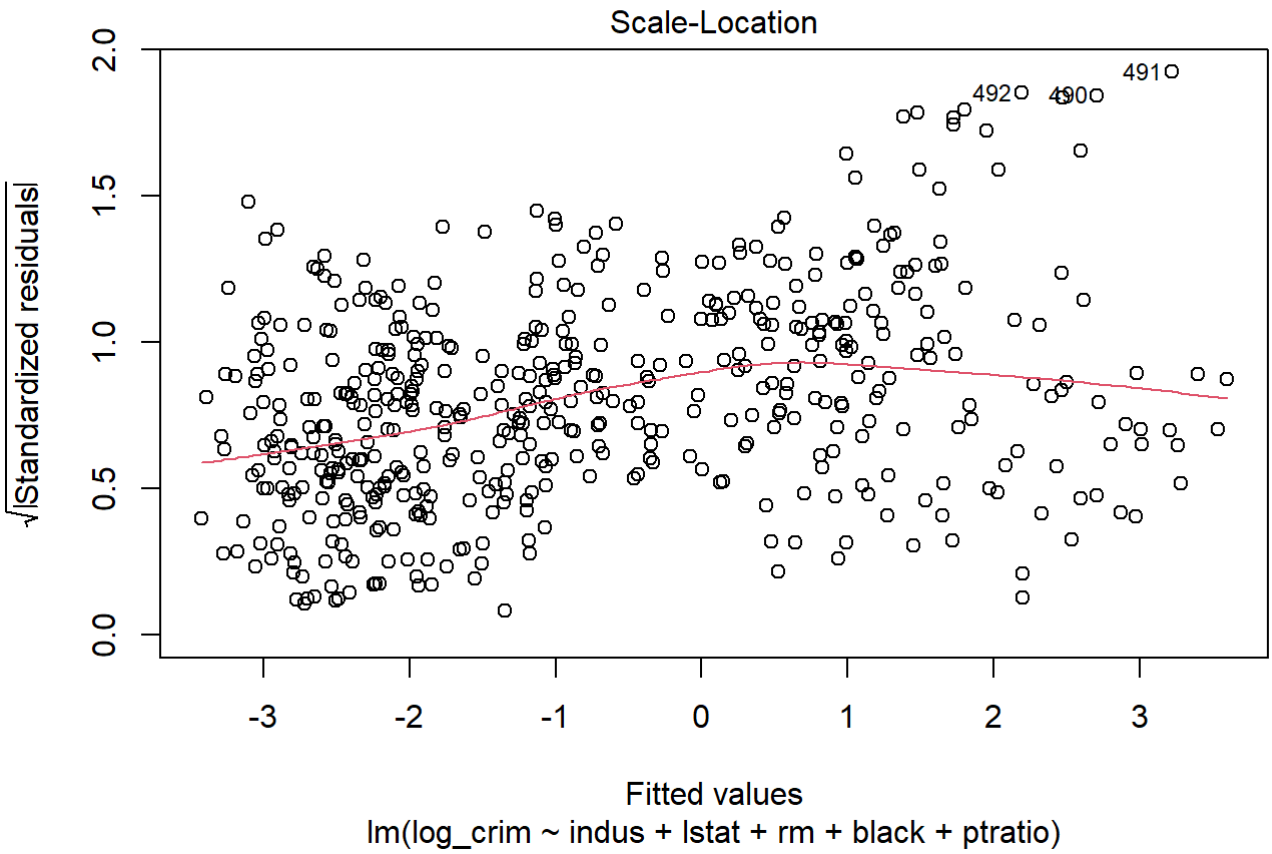
```
plot(reg_log,1)
```




```
plot(reg_log,2)
```



```
plot(reg_log,3)
```



Supprimons certaines variables

Modèle initial

```
reg <- lm(crim ~ indus + lstat + rm + black + ptratio, data = reg_data)
summary(reg)
```

```
##
## Call:
## lm(formula = crim ~ indus + lstat + rm + black + ptratio, data = reg_data)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -12.730  -2.492  -0.502   1.167  81.727
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -11.427671   5.949093  -1.921  0.05531 .
## indus         0.169012   0.061575   2.745  0.00627 **
## lstat         0.370849   0.068563   5.409 9.85e-08 ***
## rm            1.125077   0.598607   1.879  0.06076 .
## black        -0.020270   0.003916  -5.175 3.30e-07 ***
## ptratio       0.467365   0.167497   2.790  0.00547 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.274 on 500 degrees of freedom
## Multiple R-squared:  0.2919, Adjusted R-squared:  0.2848
## F-statistic: 41.22 on 5 and 500 DF,  p-value: < 2.2e-16
```

Modèle sans la variable black

```
reg_sans_indus_black <- lm(crim ~ lstat + rm + ptratio + indus, data = reg_data)
summary(reg_sans_indus_black)
```

```
##
## Call:
## lm(formula = crim ~ lstat + rm + ptratio + indus, data = reg_data)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -13.872  -2.651  -0.526   1.056  79.772
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -23.69057    5.59553  -4.234 2.73e-05 ***
## lstat         0.45631    0.06824   6.687 6.09e-11 ***
## rm            1.56987    0.60746   2.584 0.010039 *
## ptratio       0.49564    0.17166   2.887 0.004054 **
## indus         0.22602    0.06212   3.638 0.000303 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.459 on 501 degrees of freedom
## Multiple R-squared:  0.2539, Adjusted R-squared:  0.248
## F-statistic: 42.63 on 4 and 501 DF,  p-value: < 2.2e-16
```

Modèle sans les variables rm et black

```
reg_sans_black_rm <- lm(crim ~ lstat + indus + ptratio, data = reg_data)
summary(reg_sans_black_rm)
```

```
##
## Call:
## lm(formula = crim ~ lstat + indus + ptratio, data = reg_data)
##
## Residuals:
## Criminalité
##     Min      1Q  Median      3Q     Max
## -14.154  -2.788  -0.628   1.406  81.367
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -11.35223    2.93466  -3.868 0.000124 ***
## lstat         0.36968    0.05977   6.185 1.29e-09 ***
## indus         0.22654    0.06247   3.626 0.000317 ***
## ptratio       0.42075    0.17015   2.473 0.013737 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.501 on 502 degrees of freedom
## Multiple R-squared:  0.244,  Adjusted R-squared:  0.2395
## F-statistic: 54.01 on 3 and 502 DF,  p-value: < 2.2e-16
```